Annealing Markov Chain Monte Carlo with Applications to Ancestral Inference

By
Charles J. Geyer[1] and Elizabeth A. Thompson[2]

Technical Report No. 589

School of Statistics
University of Minnesota

July 1, 1993
Revised February 7, 1994

[1]Research supported in part by grant DMS-9007833 from the National Science Foundation
[2]Research supported in part by grant BIR-9305835 from the National Science Foundation and by grant GM-46255 from the National Institutes of Health

Abstract

Markov chain Monte Carlo (MCMC, the Metropolis-Hastings algorithm) has been used for many statistical problems including Bayesian inference, likelihood inference, and tests of significance. Though the method often works well, doubts about convergence remain in all applications. Here we propose MCMC methods distantly related to simulated annealing. Our samplers mix rapidly enough to be usable for problems in which other methods would require eons of computing time. They simulate realizations from a sequence of distributions, allowing the distribution being simulated to vary randomly over time. If the sequence of distributions is well chosen, the sampler will mix well and produce accurate answers for all the distributions. Even when there is only one distribution of interest, these annealing-like samplers may be the only known way to get a rapidly mixing sampler.

These methods are essential for attacking very hard problems, which arise in areas such as statistical genetics. We illustrate the methods with an application that is much harder than any problem previously done by Markov chain Monte Carlo. It involves ancestral inference on a very large genealogy (7 generations, 2024 individuals). The problem is to find, conditional on data on living individuals, the probabilities of each individual having been a carrier of cystic fibrosis. The unconditional probabilities are easy to calculate, but exact calculation of the conditional probabilities is infeasible. Moreover, a Gibbs sampler for the problem would not mix in a reasonable time, even on the fastest imaginable computers. Our annealing-like samplers have mixing times of a few hours. We also give examples of samplers for the "witch's hat" distribution and the conditional Strauss process.

The methods may also be useful for easier problems. It is a common concern about MCMC that one can never be sure that that the chain was well mixed and the answers are correct. Although we have no guaranteed convergence bounds for our methods, it does seem that annealing-like samplers are overkill in easy problems and should dispel doubts about convergence.

1. Introduction

Markov chain Monte Carlo (MCMC) in the form of the Metropolis-Hastings algorithm (Hastings 1970; Metropolis, Rosenbluth, Rosenbluth, Teller, and Teller 1953) and its special case the Gibbs sampler (Geman and Geman 1984) has been used in recent years to attack a wide variety of statistical problems that seem impossible to solve by other means. See, for example, Geyer and Thompson (1992), Geyer (1992), Besag and Green (1993), Smith and Roberts (1993), Tierney (in press) and the accompanying discussions and references. MCMC simulates realizations from probability distributions whose densities are known up to a normalizing factor. If $h(x)$ is a nonnegative integrable function on the sample space, the Metropolis-Hastings algorithm simulates a Markov chain whose equilibrium distribution is proportional to $h(x)$ using only evaluations of $h(x)$. No matter how complicated the problem, it is usually possible to find a Markov chain having the desired equilibrium distribution.

If the chain is irreducible, time averages over the chain converge to expectations with respect to the stationary distribution as the Monte Carlo sample size goes to infinity, but if the chain is slowly mixing, it may take astronomically large sample sizes to get accurate estimates. Slow mixing typically occurs in problems where the sample space has high dimension and the sampler updates one variable at a time like the Gibbs sampler. Then the mixing time can be exponential in the number of variables. As the dimension increases the Gibbs sampler becomes useless at some fairly low dimension.

To do MCMC on high-dimensional problems, better Markov chain samplers are needed, ones whose mixing time is does not increase exponentially with dimension. To do that it is necessary to make a radical change in the sampling scheme, getting away from updating one variable at a time. The first such method was the Swendsen-Wang (1987) algorithm for the Ising model and related models of statistical physics. A number of similar algorithms have been devised since (Besag and Green 1993; Wang and Swendsen 1990) and are grouped under the name "cluster algorithms." Although these algorithms are highly effective, they seem to apply only to problems where all variables are conditionally positively correlated given the rest and hence do not apply to many problems of interest to statisticians.

A much more general algorithm was proposed by Geyer (1991a) under the name "Metropolis-coupled MCMC" (MCMCMC). An improvement of MCMCMC by changing from parallel simulation of distributions at different temperatures to random temperatures, lead us to the algorithm we called "pseudo-Bayes" in the first version of this paper. We later found that the key idea had been independently proposed by Marinari and Parisi (1992) under the name "simulated tempering." We have adopted their name even though our algorithm differs from theirs in some details and adds a number of ideas needed to make it work on a wide variety of problems. This paper explains our version of simulated tempering and provides examples of its use.

Both MCMCMC and simulated tempering are based on an analogy with simulated annealing (Kirkpatrick, Gelatt, and Vecchi 1983). Simulated annealing is an algorithm for optimization rather than Monte Carlo, but it provides the useful

metaphor of starting with "heated" versions of the problem and slowly cooling down to the problem of interest. Since an MCMC algorithm is a Markov chain with stationary transition probabilities, neither MCMCMC or simulated tempering "cools" like simulated annealing, but both use a one-parameter family of probability distributions indexed by a parameter called "temperature" ranging from the distribution of interest at the "coldest" temperature to a "hottest" distribution that is much easier to simulate.

There have been other proposals in the statistics literature for speeding up the mixing of MCMC samplers, such as using the classical variance reduction methods of ordinary independent-sample Monte Carlo like importance sampling and antithetic variates (see Besag and Green 1993 and Tierney in press and the references cited therein), but those methods only reduce the mixing time by a constant factor and would not would not change the exponential growth of mixing time with dimension. There have also been other proposals in the statistical physics literature. Berg and Neuhaus (1991), following earlier work by Berg and other authors, propose to simulate a "multicanonical ensemble" as the stationary distribution of the sampler and reweight the multicanonical ensemble to the distribution of interest by the importance sampling formula. This is similar to work by Torrie and Valleau (1977) who called their importance sampling scheme "umbrella sampling," except that Torrie and Valleau do not present their method as a way of doing intractable high-dimensional problems but rather as one for obtaining stable estimates of expectations with respect to a wide range of distributions in the same spirit as the method of "reweighting mixtures" of Geyer (1991b). Frantz, Freeman, and Doll (1990) proposed a method called "J-walking" which is not an exact MCMC scheme, because it does not run a Markov chain with a specified stationary distribution, but only an approximation thereof. If it were corrected so as to be exact, it would be MCMCMC.

We provide three examples of our simulated tempering method. The "witch's hat" distribution, provides a clear illustration of how simulated tempering works when Gibbs sampling fails. A more realistic example is the Strauss process, where the method is used for importance sampling in the spirit of Torrie and Valleau (1977) and Geyer (1991b). The third example is from pedigree analysis. We analyze a small 35-member pedigree for which the exact answers are known, and large 2024-member and 5277-member pedigrees for which simulated tempering seems to be the only known feasible sampling algorithm. Although our methods were developed to do high-dimensional problems like those in pedigree analysis, they can be applied to any situation in which MCMC is used. In easier problems, these annealing-like samplers go a long way toward alleviating concerns about convergence.

2. Algorithms

Both MCMCMC and simulated tempering simulate a sequence of m distributions specified by unnormalized densities $h_i(x)$, $i = 1, \ldots, m$ on the same sample space, where the index i is called "temperature". We call $h_1(x)$ the "cold" distribution and

$h_m(x)$ the "hot" distribution. Sometimes, as in Section 3, all m distributions are of interest, but usually only the cold distribution is of interest, and the rest are used only to increase the mixing and are not of interest in themselves.

Simulated annealing uses a specific form of "heating" a problem that is sometimes called "powering up." If $h(x)$ is the unnormalized density for the distribution of interest, $h(x)^{1/\beta}$ for $\beta > 1$ are the "heated" unnormalized densities, including perhaps $\beta = \infty$ which gives $h(x) \equiv 1$. This form comes from statistical physics where the distribution of a thermodynamic equilibrium has an unnormalized density of the form $e^{-U(x)/kT}$, where $U(x)$ is the energy (Hamiltonian) function of the system, T is the absolute temperature and k is the Boltzmann constant. Such a distribution is called a Gibbs distribution and gives the Gibbs sampler its name. It is a special case of powering up with $h(x) = e^{-U(x)}$ and $\beta = kT$.

Marinari and Parisi (1992) used this kind of heating because it was natural for their example (the random field Ising model, a Gibbs distribution), and we also use it in the conditional Strauss process example (also a Gibbs distribution). However, "powering up" is *not* an essential part of simulated tempering or MCMCMC. In the "witch's hat" example, Section 2.6, "powering up" is useless, but a different form of heating works fine. It is necessary to find a form of "heating" that works well in each particular problem.

2.1. Simulated Tempering

For now, suppose that the $h_i(x)$ have been specified. Guidance for choosing them will be given after the algorithm is described. We also suppose that there is available for each i a method for updating x that has $h_i(x)$ as a stationary distribution. For example this could be a Gibbs or Metropolis update for $h_i(x)$. The state of a simulated tempering sampler is the pair (x, i) where x takes values in the common state space of all the $h_i(x)$ and the temperature i is now random. The stationary distribution of the sampler is proportional to $h_i(x)\pi(i)$ where $\pi(1)$, ..., $\pi(m)$ are auxiliary numbers that must be chosen in advance so as to make the sampler work. We call π the *pseudo-prior* because $h_i(x)\pi(i)$ looks like the product of likelihood and prior, i being the parameter and x the data, and because it determines the distribution of temperatures.

The specification of one iteration of the "Hastings version" of the simulated tempering algorithm is as follows:

1. update x using a Metropolis-Hastings or Gibbs update for h_i.

2. Set $j = i \pm 1$ according to probabilities $q_{i,j}$ where $q_{1,2} = q_{m,m-1} = 1$ and $q_{i,i+1} = q_{i,i-1} = \frac{1}{2}$ if $1 < i < m$.

3. Calculate the Hastings ratio

$$r = \frac{h_j(x)\pi(j)}{h_i(x)\pi(i)} \frac{q_{j,i}}{q_{i,j}}$$

and accept the transition (set i to j) or reject it according to the Metropolis rule: accept with probability $\min(r, 1)$.

In the calculation of r in step 3, the factor $q_{j,i}/q_{i,j}$ is the Hastings (1970) modification of the Metropolis algorithm. It compensates for asymmetry in the proposed transitions. There is also a "Metropolis version" of the algorithm which in step 2 uses the probabilities $q_{1,1} = q_{1,2} = q_{m,m-1} = q_{m,m} = \frac{1}{2}$ so the factor $q_{j,i}/q_{i,j}$ in step 3 disappears. Because half the time it does not attempt to move from $i = 1$ or $i = m$, the Metropolis version makes fewer transitions and is slightly inferior.

There are two built-in diagnostics. If any pair of adjacent distributions are too far apart, this will be indicated by low acceptance rates in step 3. Secondly, consider the *occupation numbers* of the chain, the number of iterations spent in each temperature i. If the sampler does not mix, the occupation numbers will be very uneven. This indicates the need for a better pseudo-prior. Simulated tempering has advantages over MCMCMC (Geyer, 1991a), in that we keep only one copy of the state x rather than m copies, so the chain uses less storage and also mixes better. The disadvantage is that simulated tempering needs a good pseudo-prior, which must be determined by preliminary experimentation.

2.2. Determination of the Pseudo-Prior

The stationary distribution of a simulated tempering Markov chain is a joint distribution for the pair (X, I) where X is a random realization of the state variable x and I is a random realization of the "temperature" i. The marginal distribution of I is

$$\Pr(I = i) \propto \pi(i) \int h_i(x)\, d\mu(x) = c(i)\pi(i)$$

where $c(i) = \int h_i\, d\mu$ is the normalizing constant for distribution i. Hence, if $\pi(i) = 1/c(i)$, the marginal distribution of I would be uniform, the sampler would spend a fraction $\frac{1}{m}$ of the time sampling each distribution, and there would be no temperature that is not visited frequently. The question is then how to determine the normalizing constants, since they are typically unknown. We offer three methods: (1) using an MCMCMC sampler, (2) using stochastic approximation, and (3) trial and error. In very hard problems, neither of the first two work reliably, and some experimentation is necessary. In easier problems either of the first two may be used.

If an MCMCMC sampler using the same sequence of distributions h_i as the simulated tempering sampler mixes, then a preliminary run of the MCMCMC can be used to estimate the normalizing constants, either by direct Monte Carlo integration (Geyer and Thompson, 1992; Thompson and Guo 1991) or by reverse logistic regression (Geyer 1991b).

Stochastic approximation, also called the Robbins-Munro method (Wasan 1969), for simulated tempering starts with any values for the pseudo-prior and updates the values as the chain progresses. At iteration k, the amount $c_0/[m(k + n_0)]$ is added to $\log \pi(i)$ for each i not equal to the current state I, and the amount $c_0/(k + n_0)$ is subtracted from $\log \pi(I)$. Here c_0 and n_0 are positive constants chosen by the user. It is necessary to choose a c_0 small enough and n_0 large enough so that the algorithm does not make large overcorrections early in the run before many samples

have been collected. On the other hand, if c_0 is chosen too small or n_0 too large, it will take a very long time for the algorithm to converge to a useful pseudo-prior. Stochastic approximation works well on small problems and in large problems when started from near the answer. Our experience so far is that it does not converge rapidly enough to be useful in large problems when started far from the answer.

Trial and error determines the $\pi(i)$ starting at one end of the sequence of temperatures and proceeding to the other, usually starting at the hottest temperatures, which are easier to sample. Suppose that $\pi(k+1), \ldots, \pi(m)$ have been determined so that a simulated tempering sampler for the corresponding distributions mixes well and has roughly even occupation numbers. We now want to determine a good $\pi(k)$ so that a sampler mixes when h_k is added to the set of distributions. Since $\pi(k)$ may differ from $\pi(k+1)$ by many orders of magnitude, extrapolation using the $\pi(i)$ already determined may only get within several orders of magnitude of a good $\pi(k)$. If $\pi(k)$ is set orders of magnitude too small, the chain will run down to temperature $k+1$ but never jump to temperature k. If $\pi(k)$ is set orders of magnitude too large, the chain will run down to temperature k and stay there, never jumping back up to temperature $k+1$. In either case, $\pi(k)$ is increased or decreased as appropriate by an order of magnitude or more and the sampler rerun. Once $\pi(k)$ is adjusted to the right order of magnitude, the sampler will mix going in and out of the cold distribution. Then the pseudo-prior can be more finely adjusted by dividing the $\pi(i)$ from the last run by the occupation numbers $\Pr(I = i)$ to get approximately uniform occupation numbers in the next run.

This simplest form of trial and error is very slow. It can be speeded up by using stochastic approximation. One can often extrapolate five or more new components of the pseudo-prior vector close enough for stochastic approximation to converge. Suppose that $\pi(k+5), \ldots, \pi(m)$ have been determined so that a sampler mixes. Extrapolate $\pi(k), \ldots, \pi(k+4)$. Run stochastic approximation to get the $\pi(i)$ to within an order of magnitude of the inverse normalizing constants. Then rerun the sampler without stochastic approximation to check that the sampler still mixes, and correct the pseudo-prior by dividing by the occupation numbers.

With long temperature sequences (e. g. 40 distributions), stochastic approximation even when started with a good extrapolation may fail to get close enough for the chain to mix when stochastic approximation is turned off: the $\pi(i)$ are still incorrect by more than an order of magnitude. In this case, "forcing the mixing" helps. If in step 3 of the simulated tempering algorithm, we multiply r by a constant greater than 1, this increases the acceptance rate. It also destroys the stationary distribution, but if the forcing constant is small, the difference between the forced and unforced schemes will be small, and adjustment of the pseudo-prior by dividing by the occupation numbers will be approximately right. So we start with a large forcing constant (100 or more) and reduce it in stages until the sampler mixes with no forcing.

In our experience, a useful pseudo-prior can be found in a reasonable amount of time, roughly of the same order as the time spent running the sampler once the pseudo-prior has been determined. Note that it is not necessary to have the pseudo-prior be exactly the inverse normalizing constants. A simulated tempering sampler

has the correct stationary distribution for any strictly positive pseudo-prior. It will mix faster if the pseudo-prior approximates the inverse normalizing constants fairly closely, but high precision in the approximation is not necessary.

2.3. Regeneration

Some Markov chains can be made to regenerate, and this can improve estimation (Ripley 1987). This is easily done with simulated tempering. Choose the hot distribution $h_m(x)$ so that independent sampling is possible, and when $i = m$ in step 1 of the algorithm update x with an independent sample from h_m. Given $i = m$, the next value of x does not depend on the current value, and the future path of the chain is independent of the past. The set of states (x, i) such that $i = m$ (x arbitrary) is an atom of the Markov chain, times when $i = m$ are regeneration times, and segments of the sample path between regeneration times (called *tours*) are stochastically independent.

Regeneration greatly simplifies estimation of Monte Carlo error. It also eliminates "start up bias" if we start at the atom (at temperature m) and run until another regeneration time, so the sample path consists of a number of complete tours. Let τ_k, $k = 0, \ldots, K$, with $\tau_0 = 0$ be the regeneration times. The sample path is (X_t, I_t) for $t = 1, \ldots, \tau_K$, and $I_0 = m$ (the value of X_0 is irrelevant). By an analog of Wald's lemma in sequential sampling (Nummelin 1984, pp. 76 and 81) the expectation over a complete tour is unbiased

$$E \sum_{t=\tau_{k-1}+1}^{\tau_k} g(X_t, I_t) = E\Big(g(X, I)\Big) E(\tau_1)$$

where $Eg(X, I)$ is an expectation with respect to the stationary distribution and the other two expectations are with respect to the distribution of the Markov chain.

If we are trying to determine the expectation of $f(X)$ under the cold distribution $E\Big(f(X) \;\Big|\; I = 1\Big)$, we calculate the sums

$$Z_k = \sum_{t=\tau_{k-1}+1}^{\tau_k} f(X_t) w(I_t)$$

$$N_k = \sum_{t=\tau_{k-1}+1}^{\tau_k} w(I_t)$$

for $k = 1, \ldots, K$ where $w(I)$ is 1 when $I = 1$ and 0 otherwise. Then the Z_k are i. i. d. with expectation $E\Big(f(X)w(I)\Big)E(\tau_1)$, and the N_k are i. i. d. with expectation $E\Big(w(I)\Big)E(\tau_1)$. Hence by the ergodic theorem

$$\frac{Z_1 + \cdots + Z_K}{N_1 + \cdots + N_K} \to \frac{E\Big(f(X)w(I)\Big)}{E\Big(w(I)\Big)} = E\Big(f(X) \;\Big|\; I = 1\Big) \tag{1}$$

If the variances of Z_k and N_k can be shown to be finite, the standard error of the Monte Carlo estimate can be calculated using the ratio estimator from finite

population sampling (Ripley 1987, p. 158 ff.) Let $\hat{\mu}_K$ denote the left hand side of (1) and μ the right hand side. Let $V_k = Z_k - \mu N_k$. Then the V_k are i. i. d. mean zero random variables with finite variance (say σ_V^2) that can be estimated by $\hat{\sigma}_V^2 = \frac{1}{K}\sum_{k=1}^{K} V_k^2$. Now $K^{-1/2}(V_1 + \cdots + V_K)$ is asymptotically Normal(0,σ_V^2), so

$$\sqrt{K}(\hat{\mu}_K - \mu) = \frac{\frac{1}{\sqrt{K}}(V_1 + \cdots + V_K)}{\frac{1}{K}(N_1 + \cdots + N_K)}$$

converges to Normal$(0, \sigma_V^2/\nu^2)$ where ν is the expectation of the N_k. Thus the asymptotic variance of $\hat{\mu}_K$ can be estimated by $(\hat{\sigma}_V^2/\hat{\nu}^2)/K$ where $\hat{\nu}$ is the sample mean of the N_k.

Typically only a small fraction of tours will visit the cold distribution so most of the N_k will be zero. Hence one might wonder whether it would not make more sense to average only over "informative tours" for which N_k is nonzero. It can be easily checked that one gets the same mean and variance estimates either way as long as K rather than $K - 1$ is used in computing $\hat{\sigma}_V^2$.

It is not necessary that the number of tours K be fixed in advance of the run. A simple martingale argument shows that τ_K can be any Markov stopping time, for example the first regeneration time after some fixed number of iterations (Mykland, Tierney, and Yu 1992).

Before leaving this issue we should explain a curiously attractive error. It seems natural to look at the estimates of probabilities Z_k/N_k obtained from single batches. These vary widely and seem to say something about the sampling variability, but they do not. Nothing is known about the distribution of Z_k/N_k, in particular its expectation is not the probability of interest, since $E(Z_k/N_k) \neq E(Z_k)/E(N_k)$.

The distribution of the tour lengths N_i will generally have a long tail so that there are many short tours and a few long ones that contribute most of the information. This is an unavoidable consequence of stationarity and slow mixing of the cold chain. If each tour only looks at a small region of the state space, the only way the stationary distribution can be correct is if tours that enter the cold chain in high probability regions are much longer than tours that enter in low probability regions. Any attempt to shorten the tail of the distribution of tour lengths must introduce bias.

Regeneration using an independence hot chain is not a necessary part of simulated tempering; it was not used by Marinari and Parisi (1992). However, in a hard problem where little is known about the model, it seems best to use the hottest possible distribution, that is, independent sampling. There is no way to know where it is safe to stop heating the distributions short of the "infinitely hot" independent sampling. When the sampler for the cold distribution alone would be very slowly mixing, it is actually the regeneration—excursions up to hot temperatures and back—that is providing all of the mixing. So regeneration estimates are very natural. Other variance estimation estimates may appear to be more stable, but appearances are deceiving. They cannot be better in this situation and may well be worse.

Despite this, if one knows either from theory or experience that a simulated tempering sampler without an independence hot chain mixes well and is safe to use,

then regeneration should not be used, since, all other things being equal, the fewer distributions the better. But we usually do not have such knowledge, so it seems safer to use regenerating samplers. We note that one need not have an independence hot chain to use regeneration, since regeneration could be obtained by "splitting" the hot chain (Mykland, Tierney, and Yu 1992), but we have not tried this.

2.4. How Many Distributions?

The dynamics of a simulated tempering sampler are very complex, so it is difficult to give criteria for chosing the number and spacing of the distributions. Some intuition, however, can be obtained from examining a simplified model. Consider a random walk on the integers $1, \ldots, m$ having transitions to adjacent states with probability $p/2$ and staying at the same point with probability $1-p$ for the interior points and $1-p/2$ for the endpoints. In the terminology of Feller (1968) this is a random walk with reflecting barriers at $x = 1/2$ and $x = m + 1/2$. This models a simulated tempering sampler with constant acceptance rate p independent of the state.

There are a variety of properties that could be called the mixing time of the random walk. Here we consider the expected time taken to move from one end to the other, which is the same as the "expected duration of the game" in Feller's terminology for a random walk with a reflecting barrier at $x = 1/2$ and an absorbing barrier at $x = m$. Using the methods of Feller (1968, chap. XIV) we find that the expected time to go from $x = 1$ to $x = m$ is $m(m-1)/p$.

This suggests that acceptance rates should not be too large. Certainly it is a losing proposition to double the number of distributions unless that multiplies the acceptance rate by a factor of 4. When the acceptance rate is already above 25% this is not possible. The actual sampler may behave rather differently from the random walk model, however, so we recommend acceptance rates in the range of 20 to 40%. This agrees with the behavior of some of our examples (Sections 2.7 and 4.4). It is not always possible, though, to obtain acceptance rates this low (Section 2.6) no matter how wide the temperature gaps. So this recommendation cannot apply to all models. The problem is that acceptance rates averaged over the whole sample space may not be reflective of acceptance rates in parts of the sample space that are important for mixing (Section 2.6). So average acceptance rates may not be a sufficient guide, but we have no better proposal at this time.

2.5. Adjusting the Spacing

Given information about the acceptance rates for a run, how can we adjust the number and spacing of the distributions to get a desired acceptance rate? Suppose the possible distributions have a one-parameter family of unnormalized densities h_λ, $0 \le \lambda \le 1$. Suppose the parameter values for a run were $0 = \lambda_1, \lambda_2, \ldots, \lambda_m = 1$, and suppose that the observed acceptance rates were $a_1, \ldots, a_{m-1}$. For $1 < i < m-1$ this rate a_i can be taken to be the average of the rates for transitions from λ_i to λ_{i+1} and vice versa. Because the $1 \to 2$ transitions are attempted twice as frequently

as the $2 \to 1$ transitions, they are accepted only half as often, the averages are unbalanced and it is best to define a_1 to be the average of the $2 \to 1$ transition rate and half the $1 \to 2$ transition rate. A similar definition is used for a_{m-1}.

The exact effect of an adjustment does not matter; any reasonable model will suffice, since the adjustment will need to be iterated in any case. We take as a model of the acceptance rate that the rate for transitions between h_{λ_i} and $h_{\lambda_{i+1}}$ is

$$a_i = \exp\left(-\int_{\lambda_i}^{\lambda_{i+1}} b(s)\,ds\right) \tag{2}$$

where $b(s)$ is some unknown function. We estimate $b(s)$ as a step function that is constant on the intervals between the λ_k

$$b(s) = b_i = \frac{1}{\lambda_{i+1} - \lambda_i} \log \frac{1}{a_i}, \qquad \lambda_i < s < \lambda_j$$

The following algorithm then determines new intervals with endpoints $\lambda_1^*, \lambda_2^*, \ldots$ that have a specified acceptance rate α according to the model (2).

1. Set $\lambda_1^* = 0$, and set $i = j = 1$.
2. Set $r = \alpha$.
3. Set $\lambda_{i+1}^* = \lambda_i^* + \frac{1}{b_j} \log \frac{1}{r}$.
4. If $\lambda_{i+1}^* < \lambda_{j+1}$, increase i by 1 and go to step 2.
5. Set $r = r/\exp\bigl(b_j(\lambda_{j+1} - \lambda_i^*)\bigr)$ Increase j by 1 and go to step 3.

Experience shows that this method tends to overshoot in its corrections. If the observed acceptance rates are about 90% and one asks for 30%, it may produce λ^*'s that give acceptance rates varying from 10 to 30 percent. A few iterations of the process, however, do give approximately uniform acceptance rates at the desired level.

2.6. The Witch's Hat Distribution

The "witch's hat" distribution in two dimensions is the distribution on the unit disc with a density shaped like a witch's hat, with a broad flat brim and a high conical peak. It was proposed by Matthews (1991) as a counterexample to the Gibbs sampler. In higher dimensional analogs of the two dimensional distribution, the mixing time of the Gibbs sampler increases exponentially fast with dimension, since all but one coordinate must be lined up with the peak before a Gibbs step can move from the brim to the peak and this has exponentially small probability.

Here we use for illustration a simplified witch's hat distribution defined as follows. Let α and β be real numbers with $0 < \alpha \leq 1$ and $\beta \geq 0$. Define a distribution on the unit hypercube in d dimensions $[0,1]^d$ as follows. The unnormalized density is

Table 1: Results for the simplified witch's hat distribution. The cold distribution is the top row and the hot distribution the bottom; α and β are the parameters of the witch's hat distribution, μ is the probability of the peak, which is equal to α for the β values chosen here, $\hat{\mu}$ is the estimate of μ obtained by averaging over the samples. The "actual error" is the difference between $\hat{\mu}$ and $\mu = \alpha$. The "estimated error" is the standard error of $\hat{\mu}$ estimated using the ratio estimator.

α	β	$\hat{\mu}$	*actual error*	*estimated error*
0.333	1.03×10^{14}	0.335	0.001	0.031
0.351	2.32×10^{13}	0.354	0.003	0.031
0.370	5.24×10^{12}	0.373	0.003	0.031
0.390	1.19×10^{12}	0.382	−0.008	0.030
0.411	2.70×10^{11}	0.403	−0.008	0.030
0.433	6.14×10^{10}	0.424	−0.009	0.029
0.456	1.41×10^{10}	0.441	−0.016	0.028
0.481	3.23×10^{9}	0.458	−0.023	0.027
0.507	7.45×10^{8}	0.486	−0.021	0.026
0.534	1.73×10^{8}	0.510	−0.023	0.024
0.562	4.04×10^{7}	0.541	−0.021	0.022
0.593	9.52×10^{6}	0.570	−0.023	0.021
0.624	2.27×10^{6}	0.607	−0.018	0.019
0.658	5.46×10^{5}	0.642	−0.016	0.017
0.693	1.34×10^{5}	0.676	−0.017	0.015
0.731	3.33×10^{4}	0.715	−0.016	0.013
0.770	8.55×10^{3}	0.759	−0.011	0.010
0.811	2.28×10^{3}	0.810	−0.002	0.007
0.855	6.46×10^{2}	0.855	0.000	0.005
0.901	1.99×10^{2}	0.903	0.002	0.003
0.949	6.98×10^{1}	0.948	−0.001	0.001
1.000	0.00	1.000	0.000	0.000

equal to $1 + \beta$ on the small hypercube $[0, \alpha]^d$ and equal to 1 elsewhere in $[0, 1]^d$. We still call the part of the distribution over the small hypercube the "peak" and the rest the "brim" although the density no longer looks much like a witch's hat. These distributions for various values of the parameters α and β make up the simplified witch's hat family.

For our example, we used $d = 30$ and 22 temperatures shown in Table 1. The hot distribution was the uniform distribution on the unit hypercube, which is a simplified witch's hat distribution with $\alpha = 1$ or $\beta = 0$. The cold distribution had $\alpha = 1/3$ and $\beta \approx 10^{14}$ chosen so the probability of the peak was exactly $1/3$. The α's for intermediate temperatures were chosen so that the α's were equally spaced on the log scale and the area of each peak is the same fraction (0.20816)

of the peak for the next higher temperature. Thus there is a constant proportion of proposals in the peak in attempted jumps down in temperature. The β's were chosen so that the probability of the peak was equal to α. Since the hot distribution permits independent sampling the sampler is regenerating. For this example we used a pseudo-prior that was exactly equal to the inverse normalizing constants $1/(1+\beta\alpha^d)$.

Some form of heating is necessary, but for the witch's hat "powering up" is useless. Raising the cold distribution to a power still produces a distribution with two levels, the peak and the brim, in the same positions, so powering up is the same as decreasing β while leaving α fixed. It should be clear that this makes the peak no easier to hit and so gives no improvement over ordinary Gibbs sampling. If the hot distribution has $\beta = 0$, it is a regeneration point, so regeneration methods can be used to estimate variance. The overall acceptance rates will be high, but almost all tours will stay in the brims of the distributions. Over a very long run of the sampler there will eventually be a transition from the brim to the peak of some distribution, and then the sampler will stay in the peaks for 10^{12} iterations. Until such a long tour is seen, the regeneration estimates of variance will be completely erroneous.

A Gibbs sampler for the cold distribution has a very hard time. The volume of the peak is $(1/3)^{30} = 5 \times 10^{-15}$ so it takes it a very long time to jump into the peak (and then stationarity requires that it take a very long time to jump out). A more careful analysis uses the fact that the peak is an atom so the Gibbs sampler is also regenerating. By the renewal theorem, the mean regeneration time is $1/P(\text{peak}) = 3$. The probability of leaving the peak in one Gibbs update is $q = 2 \times 10^{-14}$ so the probability of leaving in one scan is $1-(1-q)^d = 6 \times 10^{-13}$. In order that the average time for tours of all lengths be 3, the average length of tours of length greater than one must be 3.4×10^{12}. We can take this to characterize the mixing of the Gibbs sampler. It will need 10^{12} scans to get close to mixing and 10 or 100 times that to get any accuracy in the answers.

This is not surprising. The mixing time of the Gibbs sampler increases exponentially in d. The simulated tempering sampler, in contrast, needs a number of temperatures that is $O(d)$ and the mixing time is approximately quadratic in the number of temperatures (Section 2.4), so the mixing time is approximately $O(d^2)$.

The simulated tempering sampler was run to the first regeneration point after 1,000,000 iterations, which was iteration 1,000,110. This took 5 minutes and 42 seconds on a workstation that does about 1.5 million floating point operations per second. There were 42,556 tours of which all but 5567 were of length one (regenerations on consecutive iterations). The distribution of the regeneration times was skewed (of course) but not extremely long-tailed. The longest tour (11556 iterations) made up only 1 percent of the total iterations. The largest 17 tours made up 10 percent, the largest 165 made up 50 percent, the largest 773 made up 90 percent.

The simulated tempering sampler gets one significant figure accuracy in about 10^6 scans. The last three columns of Table 1 above give the estimates $\hat{\mu}$ of the probabilities of the peak of each distribution, the actual deviations of the estimates from the truth, and the estimated standard errors using the regeneration property and the ratio estimator of variance.

Table 2: Acceptance rates for the sampler for the simplified witch's hat distribution.

temperature	*going up*	*going down*
1		0.720
2	0.718	0.707
3	0.702	0.690
4	0.704	0.676
5	0.684	0.659
6	0.665	0.637
7	0.652	0.627
8	0.643	0.615
9	0.615	0.594
10	0.596	0.570
11	0.575	0.546
12	0.554	0.519
13	0.518	0.496
14	0.495	0.467
15	0.468	0.431
16	0.433	0.400
17	0.402	0.363
18	0.363	0.322
19	0.322	0.286
20	0.280	0.263
21	0.261	0.260
22	0.262	

The Gibbs sampler would need to run a million times as long as the simulated tempering sampler to have even a hope of diagnosing its own failure. If run for much less than 10^{12} iterations, the Gibbs sampler will give a completely wrong answer, either all or none of the iterations would be in the peak, depending on the starting position, and no diagnostic based on the samples would diagnose the nonconvergence. This is, of course, well known. It was the original point of the witch's hat problem, that the Gibbs sampler will do arbitrarily badly as the dimension increases.

Acceptance rates for jumps of the simulated tempering sampler are shown in Table 2. These acceptance rates are much larger than the recommendations in Section 2.4 at the cold end, but they cannot be made as small as 20 to 40%. Going down between temperatures 2 and 1, for example, the probability at stationarity of being on the brim before the jump is $1 - \alpha = .65$. When on the brim, the probability of a proposal on the brim is nearly one, giving a contribution to the overall acceptance rate of 65% for jumps down in temperature at points on the brim of both distributions. The probability of being in the peak before the jump is .35, and the probability of a proposal on in the peak is is 20.8% and most such proposals are accepted, giving a contribution to the overall acceptance rate of 7.3% for jumps

down in temperature at points in the peak of both distributions. So although there is an overall acceptance rate of 72%, only 7% of that is involved in simulating the peak of the cold distribution.

2.7. A Small Pedigree Example

This example uses a small instance of the genetics problems which are the subject of Section 4.1. The problem has 23 discrete variables, the genotypes of 23 individuals in a test pedigree from Thompson (1980). The problem is to find the conditional distribution of the genotypes given observed data (explained further in Section 4.1). The Gibbs sampler would work satisfactorily on this problem; we used it as a test of correctness of the code and algorithms. The exact distribution can be calculated by "peeling" (Cannings, Thompson, and Skolnick 1978), and this also gives the exact normalizing constants.

Stochastic approximation starting with a uniform pseudoprior on 13 distributions converged to within 5% of the ideal pseudoprior in 2,000,000 iterations. The method described in Section 2.5 was then used to select spacings of the distributions to obtain approximately equal acceptance rates. The results are shown in Table 3.

The mixing time of the sampler, reflected in the number of end-to-end excursions, is maximized at an acceptance rate of about one-third. This agrees with the analysis of Section 2.4 and with our experience with larger pedigrees. Acceptance rates above 50% actually make for a slower sampler. A fairly broad range of acceptance rates between 20 and 40 percent are close to optimal.

It took several iterations to achieve the fairly equal acceptance rates shown in Table 3. The first run using 13 distributions produced acceptance rates varying between 56 and 83 percent. The method of Section 2.5 applied to these results predicted equal acceptance rates of 24% with four distributions, but the actual

Table 3: Results on a test pedigree showing the effects of varying the number of distributions. Column labels: *dist.* number of distributions, *iter.* number of iterations (the first regeneration point after 2,000,000 iterations), *tours* number of complete tours from the cold distribution to the hot distribution and back to the cold, *ave.* average acceptance rate, *max.* maximum acceptance rate, *min.* minimum acceptance rate.

dist.	iter.	tours	ave.	max.	min.
20	2000546	2055	0.811	0.840	0.787
10	2000046	6177	0.614	0.635	0.597
9	2000047	7082	0.565	0.597	0.501
7	2000474	9131	0.431	0.471	0.393
6	2000090	10077	0.341	0.371	0.329
5	2000018	9515	0.213	0.247	0.183
4	2000111	5830	0.075	0.093	0.065

acceptance rates varied between 4 and 13%. So the method did not accurately predict acceptance rates (not surprising in view of the ad hoc nature of the model). Another application of the method predicted equal acceptance rates of 6.8%, and the actual acceptance rates varied between 6.5% and 9.3%. The model does fairly well at equalizing rates when the rates are not being changed much, and with iteration the method of Section 2.5 can equalize acceptance rates.

3. Likelihood Inference for the Strauss Process

The Strauss process (Strauss 1975) is the simplest non-Poisson spatial point process. Here we deal with the conditional Strauss process, which has realizations consisting of a fixed number of points in a bounded region. Let $t(x)$ denote the number of pairs of points (called *neighbor pairs*) separated by less than some fixed number r. A conditional Strauss process is any distribution in the exponential family with unnormalized densities $h_\theta(x) = e^{t(x)\theta}$ with respect to the "binomial process" under which the n points are uniformly distributed. Our example has 50 points in the unit torus and $r = 0.2$.

The first sampler for the conditional Strauss process was a Gibbs sampler described by Ripley (1979). A Metropolis sampler described by Geyer and Møller (in press) is much more efficient and, unlike the Gibbs sampler, can be used for both the unconditional and conditional processes. Even the Metropolis algorithm is inefficient for a process with strong dependence (large positive θ). A simulated tempering sampler is better.

The special case $\theta = 0$ is the binomial process, which can be sampled independently and is a regeneration point. As θ increases so does the expected number of neighbor pairs, and for large θ all of the points are in one small clump and the value of $t(x)$ is very near its maximum $\binom{50}{2} = 1225$ with very high probability. Preliminary runs showed that this occurs for $\theta > .16$, so we adjusted a sampler to have 9 distributions, $\theta = 0.0$, 0.0869, 0.1143, 0.1240, 0.1267, 0.1296, 0.1348, 0.1448, 0.16, with approximately equal acceptance rates ranging between 65 and 77 percent. The results are shown in Figure 1. Note that the horizontal coordinate is not θ but the distribution index. Fewer distributions and lower acceptance rates would have made for faster mixing, but Figure 1 would not have given as nice a picture of the Strauss process.

We ran for 405,677 iterations making 46,166 tours between regenerations, with 90 tours hitting the cold chain. The running time was 2 hours 23 minutes on a workstation that does about 1.5 million floating point operations per second. This one sample describes this conditional Strauss process for all values of θ between 0 and 0.16. In particular the mapping between the canonical parameter θ and the mean value parameter $\tau(\theta) = E_\theta t(X)$ can be determined by importance reweighting the sample. Let X_k, I_k denote the samples, which have unnormalized stationary density $h_{\theta_i}(x)\pi(i)$, and let

$$w_\theta(x, i) = \frac{h_\theta(x)}{h_{\theta_i}(x)\pi(i)}.$$

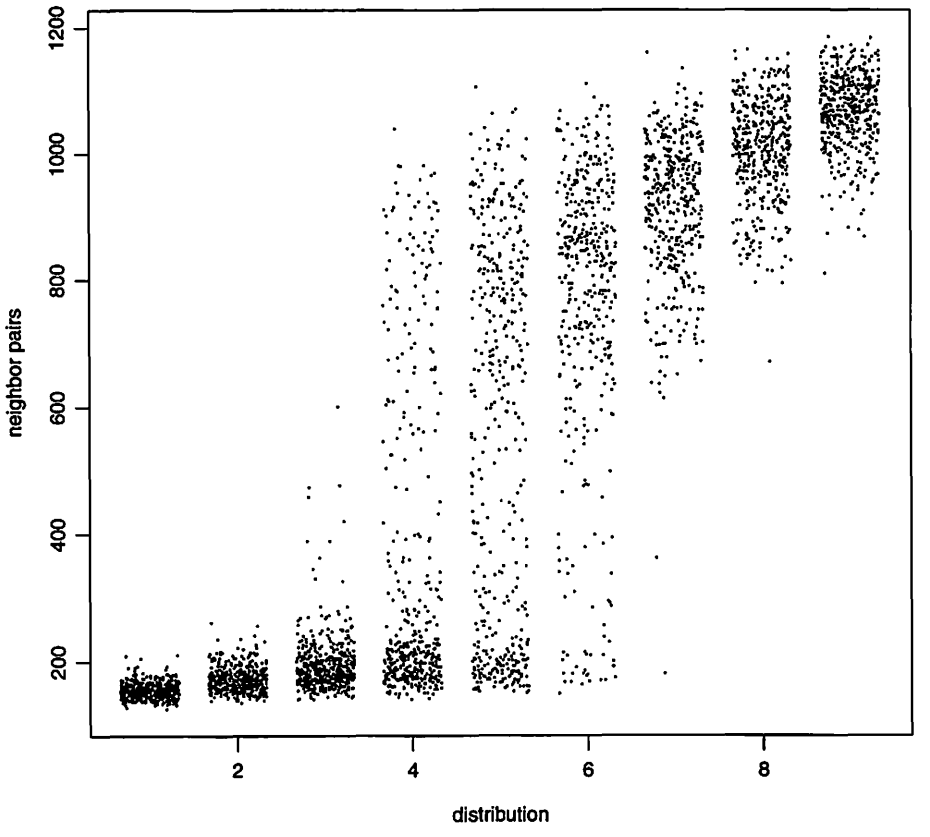

Figure 1: Scatterplot of the canonical statistic versus distribution number for the Strauss process. The x-coordinates are integer-valued, but jittered. Every 100th iteration from a run of 405,677 iterations is plotted.

Then

$$\tau_n(\theta) = \frac{\sum_{k=1}^n t(X_k) w_\theta(X_k, I_k)}{\sum_{k=1}^n w_\theta(X_k, I_k)} \to \tau(\theta), \qquad n \to \infty \tag{3}$$

for each θ and $\tau_n(\theta)$ is the natural Monte Carlo approximation of $\tau(\theta)$. This curve is shown in Figure 2. For this one-parameter exponential family maximum likelihood estimation is a simple matter of finding the θ such that $\tau_n(\theta)$ equals the observed $t(x)$. The general multiparameter case (Geyer and Thompson 1992; Geyer 1994) can be handled analogously. Monte Carlo likelihood theory applies to simulated tempering samplers just like any other Markov chain sampler. The only real novelty is in the faster mixing. To see how far the practice of Monte Carlo likelihood inference has come in just a few years, compare with Strauss (1986) in which much more computing was used to get a figure like Figure 2.

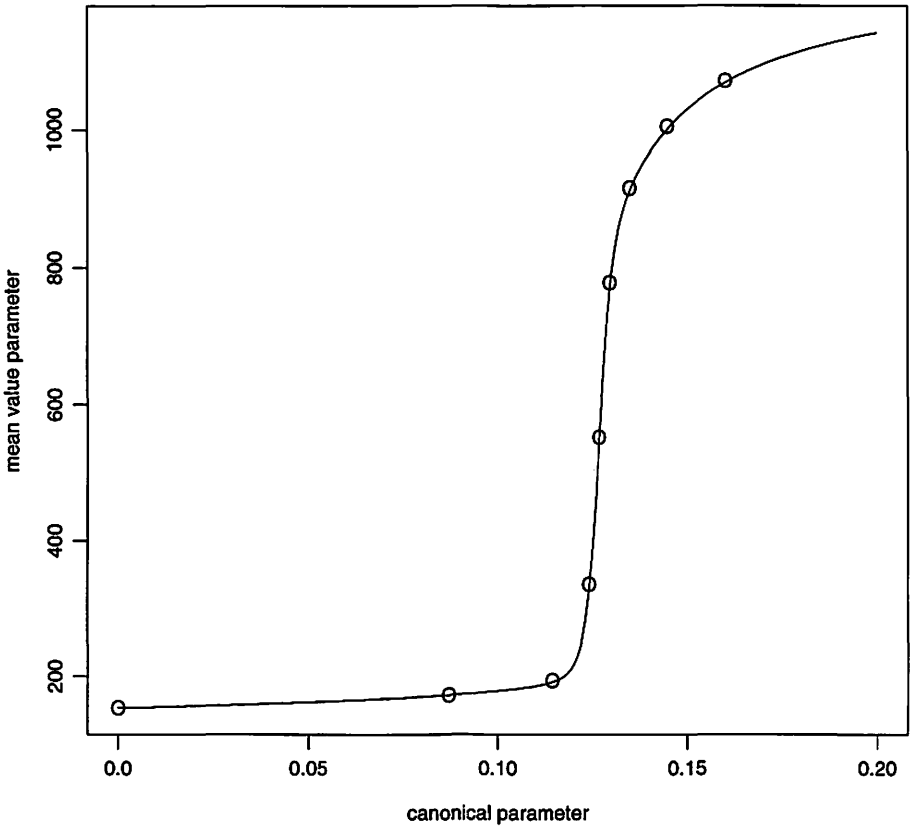

Figure 2: Plot of the mean value parameter $\tau(\theta)$ versus the canonical parameter θ for the Strauss process. The dots are the the empirical averages for the 9 distributions sampled. The line is $\tau_n(\theta)$ given by equation (3).

4. Ancestral Inference in the Hutterites

4.1. The Genetic Model

We consider the inheritance at single diallelic genetic locus. This means that each individual has two genes, one inherited from the father and one from the mother and that there are two types of genes (*alleles*) denoted A and a. Hence each individual has one of three possible genotypes AA, Aa or aa. In particular we consider a lethal recessive disease; that is, the AA and Aa genotypes produce individuals with *normal* characteristics and the aa genotype is lethal, all individuals dying before the age of reproduction. Conversely, individuals diagnosed as having the disease have genotype aa, and all individuals who have survived to adulthood (and, in particular, any parent) must be either genotype AA or Aa (called *non-carrier* or *carrier* respectively). The parents of diagnosed cases must be carriers, since they are not affected but have passed an a allele to a child. These are the *known carriers.* All other individuals have unknown carrier status. The problem of interest is to compute the probability distribution of carrier status over the pedigree given the observed data.

Mendel's laws specify the probability of an individual's genotype given the genotypes of the parents. If neither parent is a carrier, the child must be a non-carrier. If one parent is a carrier, the child has probability 0.5 of being a carrier and 0.5 of being a non-carrier. If both parents are carriers, the probability is 0.25 of the child being AA, 0.5 of being Aa, and 0.25 of being aa. Individuals whose parents are unknown (*founders*) are assumed to have genes that are a random draw from the population gene pool. Their genotype probabilities are given by

$$Pr(AA) = (1-p)^2, \quad Pr(Aa) = 2p(1-p), \quad Pr(aa) = p^2 \tag{4}$$

where p is the population frequency of the disease gene (assumed known). This specifies the probabilities in the model.

Tracing the ancestry of rare recessive diseases in genetic isolates has been often considered (for example, Castilla and Adams 1990; Hussels and Morton 1972; Sorsby 1963; Thompson and Morgan 1989). However, except where an exact probability can be computed (Thompson 1978), the methods used are of doubtful value. On a large complex pedigree, exact computation of posterior probabilities is infeasible. Although, the Gibbs sampler (Geman and Geman 1984) has been used successfully to estimate probabilities of ancestors' genotypes on small pedigrees (Sheehan 1990; Sheehan and Thomas 1993), on the pedigree of our example the Gibbs sampler does not mix, even in very long runs (M. Emond, unpublished results). For large pedigrees, methods like the Gibbs sampler that update one variable at a time can take eons to get a representative sample of genotypic configurations.

4.2. The Genealogy and Cystic Fibrosis

We illustrate the methods of this paper with a problem that has stretched them to their limits; the ancestry of cystic fibrosis (CF) genes in the Hutterite population

of North America. The structure of this large Caucasian genetic isolate has been described by Hostetler, (1974), and the CF data by Fujiwara et al. (1988). The current population of over 30,000 traces its entire ancestry to about 85 founders mostly living in the eighteenth century. About 450 immigrants came to North America in the late nineteenth century, and the population expanded very rapidly thereafter. Cystic fibrosis is a recessive and (until recently) lethal genetic disease. The frequency of CF genes in Caucasian populations is typically about 0.025; in large Caucasian populations about 1 in 1600 births is affected by CF, and about 1 in 20 individuals is a carrier. This gene frequency seems plausible for the founders of the Hutterite population, although, due to genetic drift and founder effects, the frequency in the current population may be higher.

There are 27 couples who are known to be parents of CF cases in the data set we consider (K. Morgan, personal communication). These 54 known carrier parents together with all their direct ancestors traced back to the original founders number 771. These founders, the majority of whom lived before 1750, number 77. This is the *core pedigree.*

The database of Hutterite individuals born to 1981 (T. M. Fujiwara and K. Morgan, unpublished data) contains 24,875 individuals. Analysis of this entire population pedigree is feasible, but would require huge amounts of computing time. An analysis of CF ancestry based only upon the core pedigree would, however, be biased. The ancestors of current cases had many other descendants who lived to adulthood, and therefore cannot have been affected by CF. For simplicity, we restrict attention to the offspring of members of the core pedigree. Where these offspring themselves had offspring they can be assumed unaffected. There are 1242 such individuals. Adding them to the core pedigree makes an incomplete pedigree of 2013 individuals, adding individuals so that the pedigree is "closed" (every individual has two parents or none) makes a 2024 member pedigree, which is the subject of our analysis. We later analyzed a larger pedigree of 5277 individuals, adding to the core pedigree all the children and grandchildren of the core pedigree who themselves had offspring (and so can be assumed unaffected).

In computing probabilities on pedigrees, it is often convenient to pre-process information from the periphery of the structure (Thompson 1978), and such contributions to the overall result can be incorporated into Markov chain Monte Carlo sampling on the remainder (Thompson 1991). Here, we replace children with no offspring by *pair potentials* on their parents. Let x be the genotype of such a child and x_m and x_f be the genotypes of the child's parents. Then the contribution to the probability distribution for this child is the pair potential

$$\phi(x_m, x_f) = \sum_x \Pr(\text{data on the child}|x) \Pr(x|x_m, x_f)$$

The marginal probability distribution for the remaining individuals is simply the distribution for the rest times the product of the pair potentials. For the Hutterites, this greatly decreases the amount of work the sampler must do. In our 2024-member pedigree, 1209 have no offspring in this pedigree and can be replaced by pair potentials on their parents. This leaves only 815 individuals to be sampled. The sampler

not only takes less than half the time to make one scan but is also less sticky since the potentials provide part of the distribution exactly. In the 5277-member pedigree, 3167 individuals were replaced by pair potentials leaving 2110 actually sampled.

4.3. Hot Distributions and Hot Priors

The regeneration method needs a "hot" distribution h_m for which independent sampling is feasible. For our pedigree analysis problems we used two different distributions for independent sampling: *gene drop* and *all heterozygotes*. *Gene drop* is the distribution of the genotypes when the data are ignored. It is easily simulated by drawing the founders' genotypes independently from equation (4), then going down the pedigree simulating offspring genotypes conditionally on their parents'. *All heterozygotes* is the distribution which gives probability one to the genotypic configuration in which every individual is a carrier, *Aa*. (The cases, who are known to have genotype *aa*, are not in the 2024-member or 5277-member pedigrees.) This distribution is even easier to simulate; every realization is the same. Similar distributions can be found for other problems. There is often some special case (such as no data in pedigree analysis) for which independent sampling is possible, and, when the state space is discrete, a distribution concentrated at one point can always be chosen.

There is no reason not to change other aspects of the model as well. We also experimented with individual-specific "hot priors" changing the prior distribution for certain founders so that the gene drop would make them carriers more frequently. Adjusting the hot priors so that the founders have approximately the same carrier frequencies in both the hot and cold distributions makes the sampler more efficient, but this requires some iteration. Note that the hot priors do not alter the cold distribution; the sampler mixes faster with good hot priors but it produces valid results regardless.

Either of these two hot distributions can be thought of as resulting from altering the penetrances (probability of observed data given the genotypes). The gene drop distribution results from uniform penetrance $\Pr(\text{data}|\text{genotype}) = \frac{1}{m}$ for all data values and all genotypes, where m is the number of data values, and the all heterozygotes distribution results from complete penetrance of the heterozygote genotype with data on all individuals $\Pr(\text{data}|\text{heterozygote}) = 1$ and $\Pr(\text{data}|\text{genotype}) = 0$ for the other genotypes. For "warm" distributions intermediate between hot and cold we used penetrances that were convex combinations of the hot and cold penetrances, λ of the hot penetrances and $1-\lambda$ of the cold penetrances, where $0 \leq \lambda \leq 1$. When hot priors were used, the warm distributions had similar convex combinations of the cold and hot priors.

4.4. Results

The results of our analysis of the 2024-member pedigree are shown in Figure 3 and the first two columns of Table 4. Figure 3 gives a histogram of all the carrier probabilities. These should be compared with the prior mean (unconditional proba-

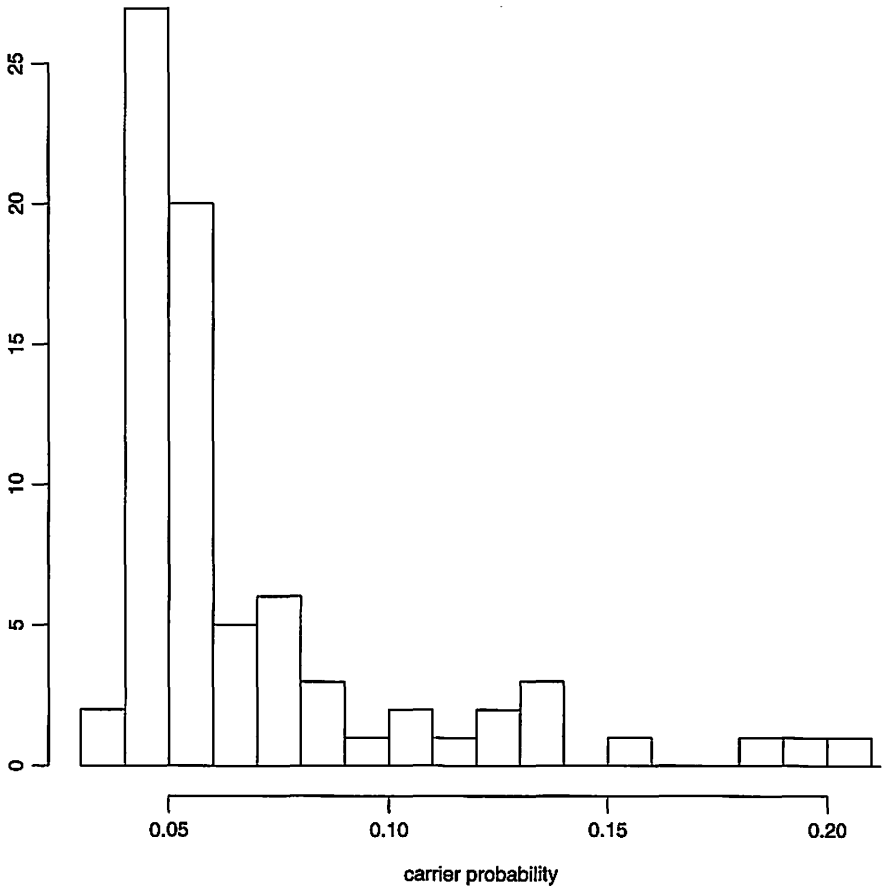

Figure 3: Histogram of the estimated carrier frequencies for the 76 founders of the core pedigree who were not known carriers.

Table 4: Hutterite Carrier Frequencies. Column labels: *2024 members* refers to the pedigree containing ancestors of affected individuals and their first generation offspring who themselves had offspring and are thus known to not have CF, *5277 members* refers to the pedigree containing ancestors of affected individuals and their first and second generation descendents who themselves had offspring, *mean* is the estimated posterior probability of being a carrier, *s. e.* is the Monte Carlo standard error of the estimate. The first column gives arbitrary labels for the individuals. The pairs C-D, E-F, G-H, and I-J are married couples with no other spouses.

	2024 members		*5277 members*	
	mean	*s. e.*	*mean*	*s. e.*
A	0.204	0.005	0.318	0.024
B	0.195	0.015	0.294	0.031
C	0.183	0.014	0.088	0.021
D	0.159	0.011	0.089	0.023
E	0.140	0.013	0.140	0.019
F	0.134	0.013	0.109	0.015
G	0.133	0.014	0.076	0.011
H	0.127	0.012	0.071	0.009
I	0.121	0.008	0.164	0.015
J	0.116	0.008	0.163	0.016
K	0.109	0.011	0.073	0.016
L	0.104	0.009	0.063	0.011
M	0.094	0.014	0.060	0.007

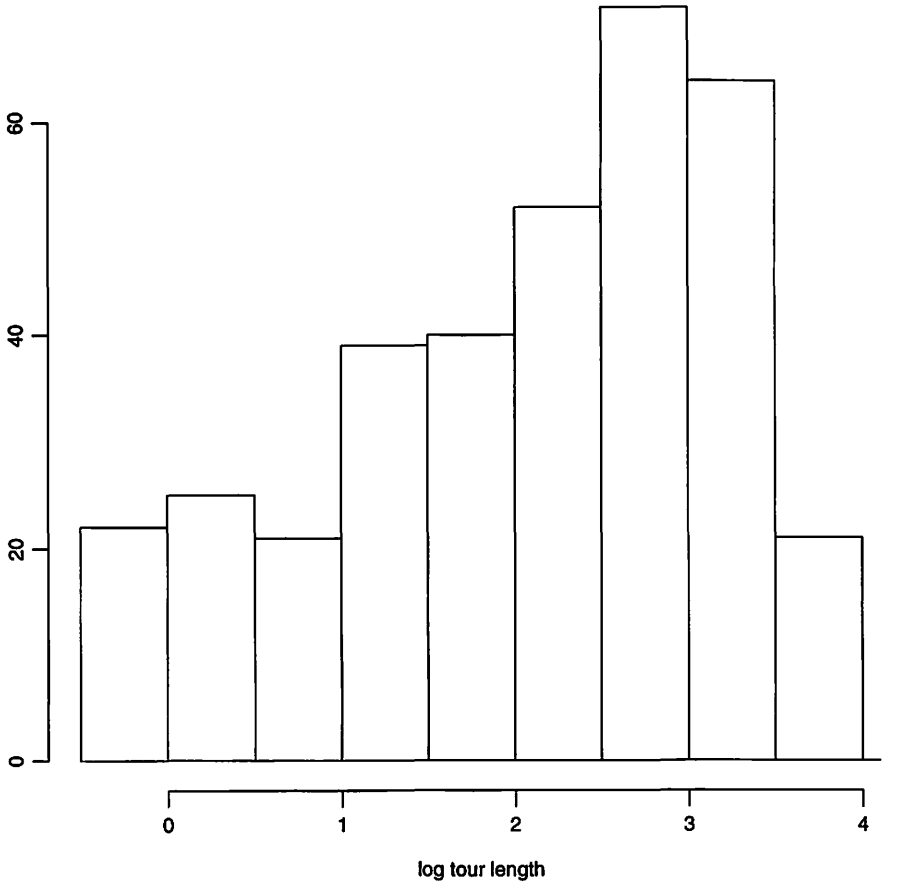

Figure 4: Histogram of log (base 10) of tour lengths in the Monte Carlo for the Hutterite pedigree.

bility) of being a carrier, which is 0.049. Of the 77 founders of the core pedigree, one is a known carrier. Of the other 76 founders, 26 are below the prior carrier probability and 50 above, though 31 of these are less than 2 standard errors (of the Monte Carlo) from the prior mean. Of the 45 founders who are more than two standard errors from the prior mean, 12 are below the prior mean and and 33 above. A few founders are far above the unconditional probability and hence are much more likely to have been carriers. The 13 with the highest carrier probabilities (as estimated by the Monte Carlo) are shown in Table 4. Their probabilities of being carriers range from almost 2 to over 4 times the prior probability. Although this is only a weak check, it is comforting that the couples C-D, E-F, G-H, and I-J, who must have exactly the same true carrier probabilities, have Monte Carlo estimates that agree to within the estimated Monte Carlo error. The conditional expectation of the number of CF genes in these 76 founders is 5.58 (standard error 0.05) as compared to the unconditional expectation of 3.705.

These estimates were based on a run of 11,555,470 iterations (each iteration being one Gibbs scan of the 815 individuals being sampled plus an attempt to jump from

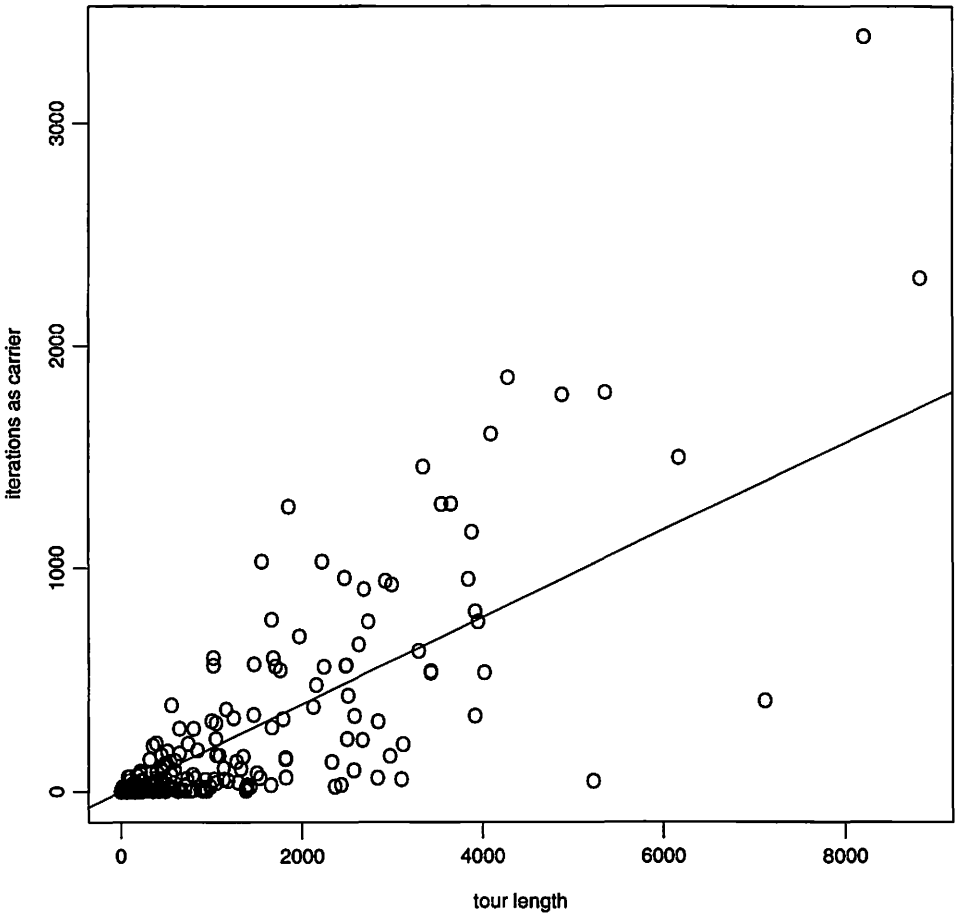

Figure 5: Scatterplot of number of iterations during a tour that individual "B" was a carrier against tour length. The line goes through the origin and has slope equal to the estimated carrier frequency for individual "B."

one distribution to another) during which there were 355 tours that spent any time sampling the distribution of interest. The total running time was 20 days 3 hours on a workstation that does about 2 million floating point operations per second.

The standard errors are based on the sampling variability of these 355 tours. The distribution of tour lengths is shown in Figure 4. The tours range in length from 1 to 8830 and approximately follow Zipf's law: 35 tours account for half of the total length, another 38 account for half of the remaining half, another 37 for half of the remaining quarter, another 34 for half of the remaining eighth, another 29 for half of the remaining sixteenth, and so forth.

The estimation for a single individual is illustrated by Figure 5, which shows the results of the Monte Carlo for individual labeled "B" in Table 4, who was chosen because he or she had high carrier probability and also large Monte Carlo error (being at the top of the pedigree). The slope of the line in the figure is the sum of all the y values of the points divided by the sum of all the x values. So the points cluster around the line in a sense, but not in any very obvious one. It is clear that

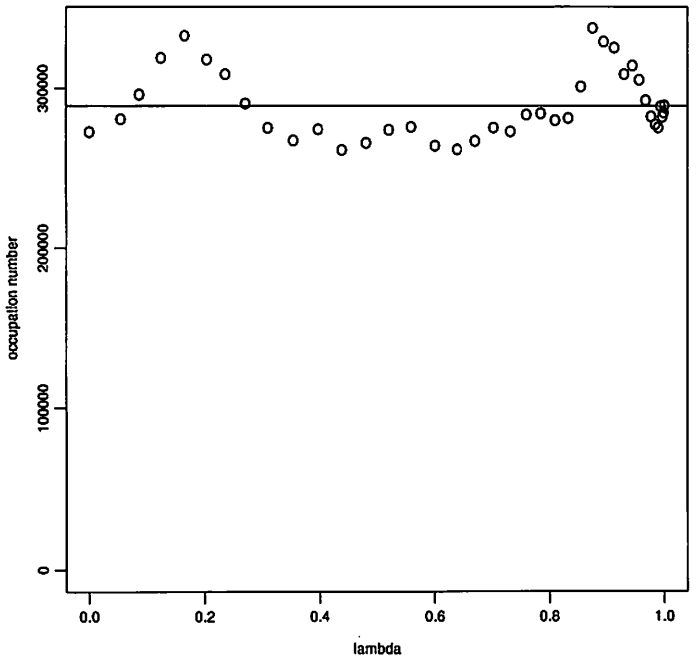

Figure 6: Occupation number for the Hutterite pedigree sampler.

the long tours provide most of the information.

The operating characteristics of the sampler are shown in Figures 6 and 7. Figure 6 shows the occupation numbers as a function of λ: the "occupation number" is the number of iterations (of the 11,555,470 total) that the sampler spent in each of the 40 distributions being sampled, and λ is the parameter indexing the distributions. The variation around the horizontal line, along which all the dots would lie if the adjustment of the pseudo-priors were perfect is mostly adjustment error, not sampling variation. So the pseudo-priors are not perfectly adjusted. They are, however, adjusted well enough so as not to degrade performance seriously. Figure 7 shows the acceptance rates. These were set using the adjustment procedure outlined in Section 2.5 with a target acceptance rate of 40 percent. As can be seen, this adjustment was not perfect either, especially at the "cold" end of the sequence of distributions, where the information from preliminary runs was least accurate. Again, the λ's are not so misadjusted as to seriously degrade performance. The main problem here is not that some of the acceptance rates deviate appreciably from 40 percent, but that it is not known whether 40 percent acceptance rates are near optimality.

Using the information from this run, both the pseudo-priors and the λ's were adjusted in an attempt to get a sampler with even acceptance rates of about 30 percent and even occupation numbers. This sampler had 32 distributions. The results of a run of 2,255,775 iterations showed that the adjustment was fairly successful. The occupation numbers were almost uniform (perhaps to within sampling error), and the acceptance rates were all 30 or 31 percent except for three steps which had rates of 29, 33 and 35. This sampler appeared to run about 8 percent faster that

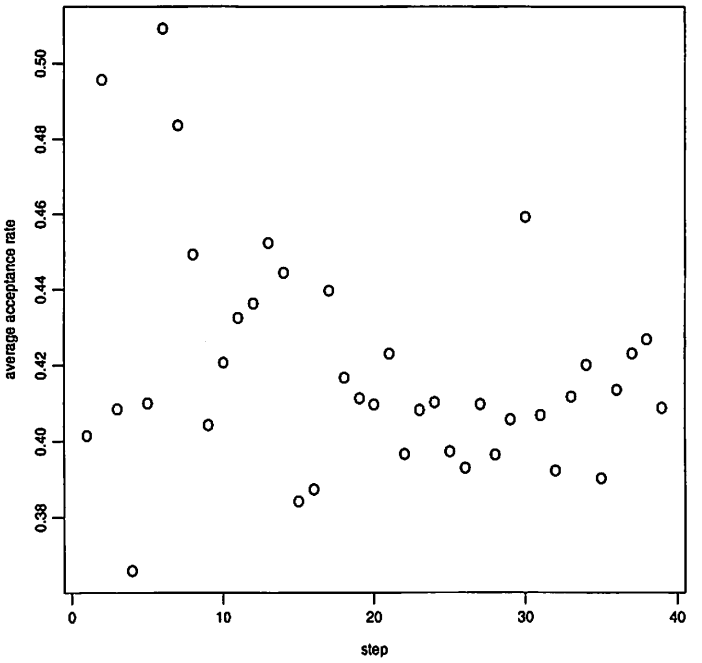

Figure 7: Plot of the average acceptance rate of jumps between distributions. The average is the average of the acceptance rate going up and the acceptance rate going down. At the ends, the acceptance rates were adjusted to account for the uneven proposal probabilities. The "steps" are numbered from 1 to 39 going from cold to hot.

the other, but the difference in speed may have been only sampling variation. After this, another sampler with 26 distributions was adjusted to have acceptance rates of about 20 percent. The results of a run of 2,008,438 iterations showed almost uniform occupation numbers and acceptance rates all between 19 and 21 percent except for three rates of 18, 22, and 26. This sampler appeared to run about 5 percent faster than the one with 30 percent acceptance rates. The sampling error in the speeds of these two samplers is fairly large, but these results do agree qualitatively with the experiment of Section 2.7: adjusting the acceptance rates to be between 20 and 40 percent seems reasonable.

Results on the 5277-member pedigree are shown in the second two columns of Table 4. This sampler had 42 distributions and ran for 12,314,658 iterations, producing 37 tours that hit the cold distribution. The tours for this sampler are about 10 times the length of tours for the 2024-member pedigree. Because of the smaller number of tours, this sampler is less accurate than the one for the 2024-member pedigree, but it is accurate enough to show that the two pedigrees do have different probability distributions. Individuals A and B are now much more likely than the others to have been carriers, half again as likely as given the information in the 2024-member pedigree. Presumably the answer for the full pedigree has the probability of A and B being carriers higher still.

5. Discussion

For the purposes of discussion let us divide problems into "hard" ones that need simulated tempering and "easy" ones for which the Gibbs sampler or variable-at-a-time Metropolis algorithms work. The main value of simulated tempering is that it provides a method of attack for these "hard" problems. The method is not guaranteed, since if one choses a bad form of "heating" simulated tempering can fail, as the example of the witch's hat with "powering up" shows (Section 2.6). But no other MCMC method has guaranteed convergence either, and simulated tempering seems to provide the best chance of obtaining a converging sampler in hard problems.

In easy problems the function of simulated tempering is to remove doubts about convergence of the Gibbs sampler and other simple methods. If simulated tempering produces the same answer as the simpler methods, then both presumably are right. There has been much controversy in the literature over the convergence even of very simple examples (Gelman and Rubin 1992; Geyer 1992). In such cases the ultimate solution should be to run simulated tempering, which seems to deliver all of the benefits that were promised for multistart methods by Gelman and Rubin (1992). Multistart methods are worthless in hard problems. Figure 5 shows why. A multistart method would produce some average over the dots in the figure that would depend on the starting distribution and hence be incorrect unless the starting distribution were very near the stationary distribution. Unless the starting distribution involved some form of annealing, the averages would be wildly incorrect.

Have we found effective hot distributions for the Hutterite CF problem? The sampler found "modes" in which each founder was a carrier, so it could have missed a mode only if the mode were characterized by some more complex function of the paths of descent of the CF genes. We used two different hot distributions. The results for the "gene drop" hot distribution have not been shown, but agreed with those discussed to within the estimated Monte Carlo error. So what evidence there is suggests we have obtained correct results. No other method we know of mixes well enough to provide a check on our results. We cannot guarantee our results are correct, but they are the best that can be done with the current state of the art.

Acknowledgements

The authors thank K. Morgan for access to data on the Hutterite genealogy. These data were compiled by T. M. Fujiwara, K. Morgan, and J. Crumley with support from by the Canadian Genetic Diseases Network. We thank Augie Kong for discussions about MCMC, in particular, his explanation that Bayesian "data augmentation" versions of Gibbs sampling are MCMCMC in time instead of space, which is suggestive of simulated tempering. We thank Peter Green for telling us about Marinari and Parisi (1992) and Neal Madras for telling us about Torrie and Valleau (1977). We thank Myles Hollander for pointing out Berg and Neuhaus (1991) and Frantz, Freeman, and Doll (1990), and the associate editor and referees

of JASA for other helpful suggestions, in particular the suggestion to do the Witch's hat example.

References

Berg, B., and Neuhaus, T. (1991), Multicanonical algorithms for first order phase transitions. *Physics Letters B*, 267, 249–253.

Besag, J., and Green, P. J. (1993), "Spatial Statistics and Bayesian Computation" (with discussion), *Journal of the Royal Statistical Society*, Ser. B, 55, 25–37.

Cannings, C., Thompson, E. A., and Skolnick, M. H. (1978), "Probability Functions on Complex Pedigrees," *Advances in Applied Probability*, 10, 26–61.

Castilla, E. E., and Adams, J. (1990), "Migration and Genetic Structure in an Isolated Population in Argentina: Aicuna," in *Convergent Issues in Genetics and Demography*, eds. J. Adams, A. Hermalin, D. Lam, and P. Smouse, Oxford University Press.

Feller, W. (1968), *An Introduction to Probability Theory and its Applications*, (Vol. 1, 3rd ed., rev.), New York: John Wiley.

Frantz, D. D., Freeman, D. L., and Doll, J. D. (1990), "Reducing Quasi-Ergodic Behavior in Monte Carlo Simulations by J-walking: Applications to Atomic Clusters. *Journal of Chemical Physics*, 93, 2769–2784.

Fujiwara, T. M., Morgan, K., Schwarz, R. H., Doherty, R. A., Miller, S. H., Klinger, K., Stanislovitis, P., Stuart, N., and Watkins, P. C. (1988), "Genealogical Analysis of Cystic Fibrosis and Chromosome 7q RFLP Haplotypes in the Hutterite Brethren," *American Journal of Human Genetics*, 44, 327–337.

Gelman, A., and Rubin, D. B. (1992), "Inference from Iterative Simulation using Multiple Sequences" (with discussion), *Statistical Science*, 7, 457–511.

Geman, S., and Geman, D. (1984), "Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 6, 721–741.

Geyer, C. J. (1991a), "Markov Chain Monte Carlo Maximum Likelihood," *Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface*, 156–163.

Geyer, C. J. (1991b) "Reweighting Monte Carlo Mixtures," Technical Report No. 568, School of Statistics, University of Minnesota.

Geyer, C. J. (1992), "Practical Markov Chain Monte Carlo" (with discussion), *Statistical Science*, 7, 437–511.

Geyer, C. J. (1994), "On the Convergence of Monte Carlo Maximum Likelihood Calculations," *Journal of the Royal Statistical Society*, Ser. B, to appear.

Geyer, C. J., and Møller, J. (in press), "Simulation and Likelihood Inference for Spatial Point Processes," *Scandinavian Journal of Statistics*.

Geyer, C. J., and Thompson, E. A. (1992), "Constrained Monte Carlo Maximum Likelihood for Dependent Data" (with discussion), *Journal of the Royal Statistical Society*, Ser. B, 54, 657–699.

Hastings, W. K. (1970), "Monte Carlo Sampling Methods Using Markov Chains and Their Applications. *Biometrika*, 57, 97–109.

Hostetler, J. A. (1974), *Hutterite Society*, Baltimore, MD: Johns Hopkins University Press.

Hussels, I. E., and Morton, N. E. (1972), "Pingelap and Mokil Atolls: Achromatopsia," *American Journal of Human Genetics*, 24, 304–309.

Kirkpatrick, S., Gelatt, C. D., Jr., and Vecchi, M. P. (1983), "Optimization by Simulated Annealing," *Science*, 220, 671–680.

Marinari, E., and Parisi G. (1992). "Simulated Tempering: A New Monte Carlo Scheme," *Europhysics Letters*, 19, 451–458.

Matthews, P. (1993), "A Slowly Mixing Markov Chain with Implications for Gibbs Sampling," *Statistics and Probability Letters*, 17, 231–236.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., and Teller, E. (1953), "Equation of State Calculations by Fast Computing Machines," *Journal of Chemical Physics*, 21, 1087–1092.

Mykland, P., Tierney, L., and Yu, B. (1992), "Regeneration in Markov Chain Samplers," Technical Report No. 585, University of Minnesota, School of Statistics.

Nummelin, E. (1984), *General Irreducible Markov Chains and Non-Negative Operators*, Cambridge University Press.

Ripley, B. D. (1979), "Simulating Spatial Patterns: Dependent Samples from a Multivariate Density," *Applied Statistics*, 28, 109–112.

Ripley, B. D. (1987), *Stochastic Simulation*, New York: John Wiley.

Smith, A. F. M., and Roberts G. O. (1993), "Bayesian Computation via the Gibbs Sampler and Related Markov Chain Monte Carlo Methods" (with discussion), *Journal of the Royal Statistical Society*, Ser. B, 55, 3–23.

Sheehan, N. (1990), "Genetic Reconstruction on Complex Pedigrees," unpublished Ph. D. dissertation, University of Washington, Dept. of Statistics.

Sheehan, N., and Thomas, A. (1993), "On the Irreducibility of a Markov Chain Defined on a Space of Genotype Configurations by a Sampling Scheme," *Biometrics*, 49, 163–175.

Sorsby, A. (1963), "Retinitis Pigmentosa in the Tristan da Cunha Islanders," *Transactions of the Royal Society of Tropical Medicine and Hygiene*, 57, 15–18.

Strauss, D. J. (1975), "A Model for Clustering," *Biometrika*, 62, 467–75.

Strauss, D. (1986), "A General Class of Models for Interaction," *SIAM Review*, 28, 513–527.

Swendsen, R. H., and Wang, J. S. (1987), "Nonuniversal Critical Dynamics in Monte Carlo Simulations," *Physical Review Letters*, 58, 86–88.

Thompson, E. A. (1978), "Ancestral inference II: The founders of Tristan da Cunha," *Annals of Human Genetics*, 42, 239–253.

Thompson, E. A. (1980), "Recursive Routines for Computations on Pedigrees," Technical Report No. 17, University of Utah, Dept. of Medical Biophysics and Computing.

Thompson, E. A. (1991), "Probabilities on Complex Pedigrees; the Gibbs Sampler Approach," *Computing Science and Statistics: Proceedings of the 23rd Symposium on the Interface*, 321–328.

Thompson, E. A., and Guo, S. W. (1991), "Evaluation of Likelihood Ratios for Complex Genetic Models," *IMA Journal of Mathematics Applied in Medicine and Biology*, 8, 149–169.

Thompson E. A., and Morgan K. (1989), "Recursive Descent Probabilities for Rare Recessive Lethals," *Annals of Human Genetics*, 53, 357–374.

Tierney, L. (in press), "Markov Chains for Exploring Posterior Distributions" (with discussion). *Annals of Statistics.*

Torrie, G. M., and Valleau, J. P. (1977), "Nonphysical Sampling Distributions in Monte Carlo Free-Energy Estimation: Umbrella Sampling," *Journal of Computational Physics*, 23, 187–199.

Wang, J. S., and Swendsen, R. H. (1990), "Cluster Monte Carlo Algorithms," *Physica A*, 167, 565–579.

Wasan, M. T. (1969), *Stochastic Approximation*, Cambridge University Press.